

**DEBRECENI EGYETEM  
INFORMATIKAI KAR**



**A FORMÁLIS NYELVEK TANÍTÁSA  
KÖZÉPISKOLÁBAN**

**TÉMAVEZETŐ:**

**Dr. PAPP ZOLTÁN**

**KÉSZÍTETTE:**

**AZARI HENRIETTE**

## Tartalomjegyzék

<i>Bevezetés .....</i>	<i>2</i>
Formális nyelvek témakör helye a középiskolás tantervben .....	2
Részlet a berlini tanmenetből .....	5
<i>Elméleti alapok .....</i>	<i>11</i>
A formális nyelvek kialakulása .....	11
A formális nyelvek definíciója .....	12
Formális rendszerek .....	15
Markov algoritmus .....	16
Generatív rendszerek .....	16
Generatív grammatikák .....	17
Chomsky nyelvosztályok .....	18
Nyelvek és automaták .....	20
<i>Példák, feladatok .....</i>	<i>24</i>
Alapfogalmak .....	24
Formális rendszerek .....	24
Markov algoritmus .....	25
Generatív rendszerek .....	27
Generatív grammatikák .....	27
Nyelvosztályok .....	29
Automaták .....	29
<i>Feladatok megoldásai .....</i>	<i>30</i>
<i>A formális nyelvek tanításának dokumentumai .....</i>	<i>37</i>
Formális nyelvek témakör elhelyezése a gimnáziumi tantervben .....	37
Ajánlás az éves óraszám felosztására .....	39
Óravázlat minta .....	43
<i>Tesztminta .....</i>	<i>46</i>
<i>Irodalomjegyzék .....</i>	<i>47</i>
<i>A témával kapcsolatos Internet címek .....</i>	<i>48</i>
<i>Függelék .....</i>	<i>49</i>
<i>Összefoglalás .....</i>	<i>52</i>
<i>Köszönetnyilvánítás .....</i>	<i>53</i>

## ***Bevezetés***

Napjainkban informatika tudományának és eszközrendszerének hihetetlen mértékű fejlődését az informatika oktatásának is hasonló mértékben kell követnie. Már nem elég csak korszerű alkalmazói ismereteket adni tanulóinknak, nagy hangsúlyt kell fektetni a gondolkodásfejlesztésre és az új ismeretek befogadására, újfajta szemléletmód kialakítására kell nyitottá tenni tanulóinkat.

Az emelt szintű és a fakultatív keretben történő informatikai nevelés célja a logikus gondolkodás, a problémamegoldás készségének általános fejlesztése mellett az informatikai szemlélet-, gondolkodásmód és munkamódszer kialakítása, elsajátítása, az igényességre, a pontosságra való törekvés alapvető követelményének kielégítése. Praktikus alkalmazói és szakmai tudás, készség és képesség fejlesztésével, korszerű (számítógépes) informatikai eszközök alkalmazásával felkészítés, felkészülés az informatika alkalmazásaira, a szakmai munkára, a mindennapi és a szakmai életben várható széleskörű alkalmazásokra. Célja olyan attitűd kialakítása, amely motiválttá és alkalmassá teszi az egyént az önálló számítógépes problémamegoldásra, az információs társadalom alakítására.

Ezen a gyorsan változó, fejlődő területen különösen fontos, hogy a tanulóknál kialakuljon informatikai ismereteik folyamatos megújításának az igénye.

## **Formális nyelvek témakör helye a középiskolás tantervben**

Emelt szintű informatika fakultáció keretein belül, de akár normál tanterv szerinti csoportoknál is, ahol az alsóbb évfolyamokon minőségi munka folyt, és a tanulók biztos tudást kaptak, ott a tanulók felkészültségéből, érdeklődéséből, és a továbbtanulási irányultságából adódóan szívesen foglalkoznak az informatika és a számítástudomány érdekes, és számukra új témáival is.

Lehetnek középiskolában olyan tehetséggondozó kis csoportok, önképzőkörök, ahol nem csak a középiskolás anyag elsajátítása a cél, hanem az ezen túlmutató ismeretek megszerzése is.

Ezekben az oktatási környezetekben lehet létjogosultsága a formális nyelvek alapjaival foglalkozó témakörnek. Tervezem, hogy egy nyolcosztályos gimnáziumban kísérletképpen letanítom ezt a néhány órányi anyagot, és a tapasztalataimmal majd később kiegészítem ezt a dolgozatot.

A formális nyelvekkel foglalkozó témakör feldolgozása az informatika oktatásának céljai közül az alábbiak megvalósulásában segíthet:

*az algoritmikus gondolkodás fejlesztése:* a tanulók önálló, rendszerezett, logikus gondolkodásának fejlesztése; a logikus gondolkodás lényeges a problémamegoldásban;

*kreativitás fejlesztése:* a formális nyelvek újszerű szemléletmódja, problémák, összefüggések, feladatok megértése, megoldása a legjobb kreativitásfejlesztő.

*önálló munkára nevelés és tehetséggondozás:* a témakör anyaga lehetőséget ad a tanulóknak egyéni ütemben, egyéni érdeklődés és képességek szerint munkát végezni; a témakör feladatai fejlesztik a kreativitást, ösztönzik őket a minél mélyebb megismerésre;

*együttműködésre nevelés, csoportmunka:* a feladatok megoldása elősegíti a kollektív tanulást, a problémák közös feldolgozását, a másokkal való kapcsolattartást, tervszerű, összehangolt munkát;

*az informatika társadalomban játszott szerepének felismertetése:* az informatika rohamos fejlődése az egész társadalmat gyökeresen átalakítja, s ebben az állandóan változó világban csak az érezheti otthon magát, aki érti a változásokat s azok mozgatóit; a különböző tudományágak és az informatika kapcsolatának felfedezése;

*felkészítés az információs társadalom kihívásainak fogadására:* új ismereteket befogadni képes, nyitott gondolkodású tanulók nevelése.

## Német tapasztalat

Magyarországon nincs még példa arra, hogy formális nyelveket oktassanak középiskolában. Németország ebben előttünk jár.

Németország oktatási rendszere, tanügyi struktúrája jelentősen eltér a Magyarországon tapasztalhatóktól, mivel az ország területe jelentősen nagyobb, mint hazánké. Nálunk úgy mondhatnánk centralizált az oktatás, azaz a fővárosból irányítják elsődlegesen a tanügyet, az Oktatási Minisztériumból. Németországban a teljes államot felosztották tizenhat tartományra, melyek gyakorlatilag államokat képeznek az államban. A tartományok oktatásának irányítása tartományi keretek között zajlik. Így a különböző tartományokban különbözőek a tantervek. Nem csak a szakmai középiskolákban, hanem a normál gimnáziumokban is a középiskolai tanterv része a formális nyelvek Németország több tartományában is.

**Berlin:** A német fővárosban egységes tanterv található a gimnáziumokra, szakközépiskolákra, szakgimnáziumokra, kollégiumokra és esti gimnáziumokra nézve. Igaz ezeket két részre bontották. Hetedikétől a tizedik osztályig általánosságban tanulnak informatikát és programozást, de nem kötelezően, hiszen nagyon sok helyen a diákok (szülők) választhatják, hogy szeretnék-e tanulni ilyet. Aztán a magasabb évfolyamokban már kötelezővé válik az informatika. Azok, akik az alsóbb osztályokban nem tanultak, azok azt az anyagot veszik, amit a többiek korábban, akiknek pedig van már tudásbázisuk, magasabb szintre lépnek az informatikába, azon belül a programozásban. Ilyen osztályokban tanítják **a formális nyelveket és automatákat**. Mivel ez a témakör összekapcsolódik a robotikával, ezért sok iskolában oktatják a **LEGO MINDSTORM** nevű programozási „nyelvet”.



## Részlet a berlini tanmenetből

### 4.4 Sprachen und Automaten – Nyelvek és automaták

- Vergleich natürlicher und formaler Sprachen
- Syntax und Semantik (Syntaxdiagramme) – Szintaxis és szemantika (szintaxis diagramm)
- Zustandsorientierte Modellierung
- Endliche Automaten – Véges automaták

*Leistungskursfach – tantárgyi teljesítés*

- Grammatiken und formale Sprachen – Nyelvtanok és formális nyelvek
- TURINGmaschine oder Registermaschine –Turing gépek vagy regisztergépek

#### **Kompetenzerwerb im Themenfeld – Kompetenciák a témában**

Dieses Themenfeld dient der Vernetzung zum Kompetenzerwerb in den natürlichen Sprachen.

Ez a téma-terület a természetes nyelvekben a kompetenciákhoz való kapcsolódást szolgálja.

Es werden die Gemeinsamkeiten und Unterschiede von natürlichen und künstlichen

Sprachen herausgearbeitet und bei den Schülerinnen und Schülern wird ein breites Verständnis für Sprachen entwickelt.

A természetes és mesterséges nyelvek azonosságainak és különbségeinek kidolgozása, ezáltal a tanulók jobban megértik a nyelveket.

Jede Sprache – ob natürlich oder künstlich – dient zur Kommunikation und genügt gewissen Regeln zur Bildung von Wörtern und Sätzen.

Minden nyelv – akár természetes vagy mesterséges- a kommunikációt szolgálja és kielégít bizonyos szó- és mondatképzési szabályokat.

In der Informatik werden Sprachen durch Grammatiken formalisiert.

Az informatikában a nyelvek a nyelvtanokon keresztül formalizálódnak.

Nur was sich mit den Mitteln einer formalen Sprache ausdrücken lässt, kann durch einen Computer bearbeitet werden.

Ami egy formális nyelv eszközeivel kifejezhető, csak azt képes a számítógép értelmezni.

Die Schülerinnen und Schüler verstehen die zur Problemlösung eingesetzten Programmiersprachen als spezielle formale Sprachen, die es ihnen erlauben, Probleme mit den Methoden der Informatik zu lösen.

A tanulók megértik, hogy a problémamegoldáshoz alkalmazott programnyelvek speciális formális nyelvek, amelyek lehetővé teszik számukra a problémák informatikai módszerekkel való megoldását.

Durch die Einführung des Automatenmodells vertiefen die Schülerinnen und Schüler ihr Verständnis von Informatiksystemen.

Az automatamodellek bevezetésén keresztül elmélyül a tanulók ismerete az informatika rendszereiről.

Automaten sind in verschiedenen Ausprägungen Teil der Lebenswelt der Schülerinnen und Schüler.

Az automaták különböző példákat hoznak a tanulók életéből.

Die Merkmale und Eigenschaften der im Alltag gefundenen Beispiele werden im Unterricht präzisiert, um die Automaten und die von ihnen akzeptierte formale Sprache zu thematisieren.

A hétköznapi életből vett példák jellegzetességeit és tulajdonságait mutatja be az oktatás, hogy az automaták és a velük kapcsolatos formális nyelvek oktatási témaként megfelelhessenek.

Dabei benötigt man eine Methode, um die zeitliche Abfolge von Zuständen beschreiben zu können.

Szükséges ehhez egy módszer, hogy az időbeli sorrend és állapot leírható legyen.

Dies führt zur zustandsorientierten Modellierung.

Ez vezet az állapotorientált modellezéshez.

Diese Vorgehensweise bietet den Schülerinnen und Schülern zugleich einen Zugang zur abstrakten Modellierung komplexer Systeme.

Ez az eljárás a tanulóknak a komplex rendszerek absztrakt modellezéséhez kínál módszert.

### **Mögliche Kontext – Lehetséges összefüggések**

- Robotik - Robotika
- Implementierung von Automaten in Programmiersprachen - Automaták implementációja a programozási nyelvekben
- Kellerautomat - Veremautomata
- CHOMSKY-Hierarchie - Chomsky nyelvosztályok

### **Mecklenburg – Elő-Pomeránia**

Informatikaoktatási struktúrája hasonlatos a berlinihez, hiszen itt is egy egységes tanterv áll rendelkezésre a középiskolásoknak, és ezt osztották szét ötödik és hatodik osztályosokra, hetedikről tizedik osztályig, valamint az utolsó két évfolyamra. Az első két osztályban (ötödik és hatodik évfolyamban) csak az informatika alapjaival ismerkednek a diákok, a következő

négy osztályban vagy erre építve vagy új ismeretként elsajátítják az algoritmizálás lépéseit és megismerkedhetnek a programozási nyelvekkel, mely a legtöbb esetben a **LEGO MINDSTORM**, hiszen a tizenegyedik és tizenkettedik évfolyamban már formalizálnak, mivel **ajánlott a formális nyelvek és automaták témakörének elsajátítása.**

### **Saar-vidék**

Az ország második legalacsonyabb népességű, és legkülönlegesebb nevű tartománya, melynek szintén eléggé egységes és jól strukturált tanterve van. Az informatikát alsóbb évfolyamokon választott jelleggel tanulhatják, a négyosztályos gimnáziumokban kötelezővé teszik, méghozzá két évig. A felbontása négy félévre szól. A programozás az első három félévben aktuális, illetve a harmadik és negyedik félévben már inkább formális nyelvek és automaták az úr.

### **Részlet a saar-vidéki tanmenetből**

#### **Automaten und formale Sprachen 15 Stunden**

##### **Endliche Automaten mit und ohne Ausgabe**

##### **Véges automaták kimenettel és kimenet nélkül**

Beschreibung der Funktionsweise durch Übergangstabellen, Ausgabetafeln  
(endliche Automat mit Ausgabe) und Übergangsgraphen

A működésimód leírása állapotátmenet-táblázatokkal, kimenetítáblázatokkal (véges automaták kimenettel) és gráfokkal (állapotátmenetgráfokkal).

Formale Definition endlicher Automaten als 5-Tupel

A véges automaták formális definíciója (5 soros tábla)

Konstruktion endlicher Automaten mit / ohne Ausgabe zu vorgegebenen Problemstellungen

Véges automaták szerkezete kimenettel/kimenet nélkül az előállított problémához

Deterministische und nichtdeterministische endliche Automaten

Determinisztikus és nemdeterminisztikus véges automaták

#### **Formale Sprachen - Formális nyelvek**

Definition -Definíciók

Beschreibung durch -leírás

- Aufzählung der Worte -a szavak felsorolása által

- charakterisierende Eigenschaften der Worte -a szavak tulajdonságainak jellemzése által



## **Grammatiken -Nyelvtanok**

Komponenten einer Grammatik:

Egy nyelvtan elemei:

Terminalsymbole, Variablen, Startvariable, Produktionsregeln

Terminális szimbólumok, változók, bemeneti változók, produkciós szabályok

Beispiele von Grammatiken und der zugehörigen formalen Sprachen

Példák a nyelvtanokra és a hozzá tartozó formális nyelvekre

Wortprüfung durch Ableitung

Szóvizsgálat levezetéssel

Endliche Automaten zur Modellierung von Abläufen und zur Spezifikation des Verhaltens von Maschinen können an Beispielen aus der Praxis (Getränkeautomat, Parkscheinautomat) erklärt werden.

Véges automaták modellezése a lefutástól a specifikációig, a gépek viselkedése példákon keresztül (italautomaták, parkolóautomaták)

Diese und weitere Beispiele aus der Steuerungstechnik führen zur Einführung der endlichen Automaten mit Ausgabe.

Ezek és további példák az irányítástechnikából vezetnek a kimenettel rendelkező véges automaták bevezetéséhez.

Der endliche Akzeptor ist ein wichtiger Sonderfall endlicher Automaten.

A véges elfogadóautomata egy különleges esete a véges automatáknak.

An den Beispielen eines Erzeugers für Paritätsbits und eines Prüfers für

Paritätsbits kann die Funktionsweise der endlichen Automaten mit Ausgabe und der endlichen Akzeptoren verdeutlicht werden.

A következő példákban egy generátor paritásbittel és egy ellenőrző paritásbittel működő véges kimenettel rendelkező és véges elfogadó automatát mutatunk be.

Die Konstruktion endlicher Akzeptoren zu Problemstellungen wie der Suche vorgegebener Teilstrings in Zeichenketten (pattern-matching) führt zum Problem nichtdeterministischer Automaten.

A véges elfogadó automaták szerkezete mint problémafelvetés az adott rész-string a jelláncban (pattern-matching) vezet a nemdeterminisztikus automaták problémájához.

Endliche Automaten, die Folgen von Zeichen über einem Zeichenalphabet akzeptieren, führen zur Einführung des Begriffs formale Sprache.

A véges automaták, amelyek jelek sorozatát ( egy ABC szerint) elfogadnak, vezetnek a formális nyelvek fogalmának bevezetéséhez.

Im Gegensatz zur Menge der Worte einer natürlichen Sprache können viele formale Sprachen über die Beschreibung kennzeichnender Eigenschaften der Wörter oder über Bildungsregeln festgelegt werden.

Ellentétben egy természetes nyelv szavainak sokaságával, a formális nyelvek a szavakat vagy tulajdonságokkal, vagy szabályokkal tudják leírni.

Für viele formale Sprachen ist eine vollständige Beschreibung durch charakterisierende Eigenschaften der Wörter nicht möglich.

Sok formális nyelv számára nem lehetséges (elégséges) egy állandó leírás a szavak jellemző tulajdonságain keresztül.

Grammatiken sind Regelwerke, welche die Herleitung aller Wörter einer Sprache ermöglichen.

A nyelvtanok szabálygyűjtemények, amelyekkel lehetséges egy nyelv minden szavának levezetése.

Der Zusammenhang zwischen Grammatiken und Sprachen wird an einfachen Beispielen (regulärer und kontextfreier Sprachen) erarbeitet.

A nyelvtanok és nyelvek közötti összhang egyszerű példákon keresztül (reguláris és környezetfüggetlen nyelvek) feldolgozva.

Von wesentlicher Bedeutung ist hierbei die Klärung der Frage, ob eine vorgegebene Folge von Terminalsymbolen nach den Produktionsregeln einer Grammatik aus der Startvariablen abgeleitet werden kann.

## **Automaten und formale Sprachen (Fortsetzung) 15 Stunden -Automaták és formális nyelvek (folytatás) 15 óra**

### **Reguläre Sprachen als Sprachen endlicher**

#### **Automaten / Akzeptoren**

#### **Reguláris nyelvek, mint a véges automaták/elfogadók nyelve**

Beschreibung durch

- reguläre Grammatiken
- reguläre Ausdrücke

Leírás reguláris nyelvtanok, reguláris kifejezések által.

Äquivalenz regulärer Sprachen und endlicher Akzeptoren

Reguláris nyelvek és véges elfogadó automaták ekvivalenciái.

Konstruktion endlicher Akzeptoren zu regulären Sprachen, die beschrieben sind durch

- reguläre Grammatiken
- reguläre Ausdrücke

Véges elfogadó automaták, reguláris nyelvek szerkezetének leírása

- reguláris nyelvtanokkal
- reguláris kifejezésekkel

Beispiele nichtregulärer Sprachen

Példák nemreguláris nyelvekre

Die zur Sprache eines endlichen Akzeptors gehörenden Worte werden stets nach einfachen Regeln (Konkatenation, Auswahl, Iteration) aus dem zugrunde liegenden Alphabet gebildet.

Egy véges automata által elfogadott nyelv szavai mindig egyértelmű szabályok szerint (konkatenációk, kiválasztás, ismétlés) épülnek fel az alapul szolgáló ABCből.

Diese Regeln finden sich sowohl in den Bildungsgesetzen regulärer Ausdrücke als auch in den Produktionsregeln regulärer Grammatiken wieder.

Ezek a szabályok megtalálhatók ugyanúgy a reguláris kifejezések képzési szabályaiban, mint a reguláris nyelvtanok produkciósabályaiban.

Jede durch einen endlichen Automaten festgelegte Sprache kann durch einen regulären Ausdruck oder eine reguläre Grammatik beschrieben werden und umgekehrt.

Minden véges automata által lefektetett nyelv leírható egy reguláris kifejezés vagy egy reguláris nyelvtan által és fordítva.

Zu vorgegebenen regulären Ausdrücken oder regulären Grammatiken kann nach einem einfach auszuführenden Verfahren der zugehörige endliche Akzeptor konstruiert werden.

Allerdings führt dieses Verfahren in vielen Fällen zu nichtdeterministischen endlichen Akzeptoren.

Am Beispiel der Sprache  $L = \{a_n b_n\}$  oder der Sprache der Palindrome werden die Grenzen endlicher Akzeptoren deutlich.

## **Literatur**

Rolf Socher: Theoretische Grundlagen der Informatik, fV Leipzig

Kastens, Büning: Modellierung; Hanser Verlag

Materialien zu Kara, Exorciser: Schweizer

Bildungsserver swissedu.ch

Ezekben a középiskolákban tehát már komoly tapasztalat van a formális nyelvek oktatásával kapcsolatban. A későbbiekben, ha lehetőségem lesz próbaképp letanítani ezt az anyagot, szeretném majd a saját észrevételeimet összehasonlítani a német gyakorlattal, és utólag ezzel is kiegészíteni a dolgozatomat.

## ***Elméleti alapok***

### **Bach Iván: Formális nyelvek c. egyetemi tankönyve alapján**

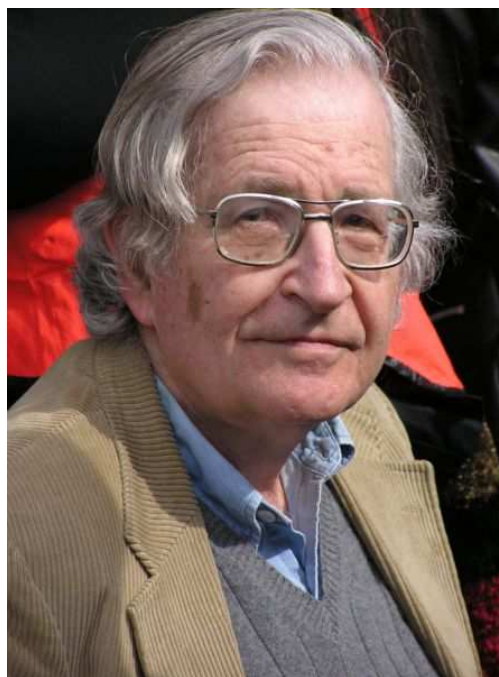
A formális leírás azt jelenti, hogy valamilyen valós, vagy absztrakt rendszert a benne lévő összefüggésekkel együtt jelekkel ( betűk , számok, egyéb szimbólumok) leírunk úgy, hogy matematikai szempontból kezelhetőek legyenek. Ennek egyik fő alkalmazása a formális nyelvészet.

### **A formális nyelvek kialakulása**

A második világháború alatt és után megpezsdült a tudományos élet, a kutatási és fejlesztési munkákban forradalmi változások következtek be. Megnőtt a magasan képzett szakemberek száma, egyre több publikáció, folyóirat, könyv jelent meg. Az angolon kívül megjelentek az orosz, japán, vagy akár magyar nyelven írt tudományos munkák. Az információhoz való hozzáférés igénye hirtelen megnövelte a fordítandó cikkek számát.

Ebben az időben kezdett kiteljesedni egy új tudomány, a számítástechnika, ami viszont azt állította, hogy célja a gépies szellemi munka kiváltása. Magától értetődő volt tehát az a törekvés, hogy a fordítást, ezt a gépies szellemi munkát egy számítástechnikai eszköz, a számítógép végezze.

A számítógép azonban csak szabatosan megfogalmazott feladatok megoldására képes, tehát szükség volt a fordítás, mint feladat formális leírására. Ekkor, az 50-es években született meg szinte a semmiből egy új tudomány, a matematikai nyelvészet. Születési évének 1956-ot tekinthetjük, ekkor publikálta a szakma atyja Noam Chomsky munkásságának első eredményeit.



A természetes nyelveknél sokkal egyszerűbbek a mesterséges nyelvek, nevezetesen azok, amelyeket számítógépek programozására használnak. Fogalmi körük erősen leszűkített, szókincsük szegényes, nyelvtani szabályaik egyszerűek, és nem ismernek kivételeket.

A matematikai nyelvészet eredményei tehát jóval könnyebben voltak alkalmazhatóak a gépi nyelvek fordításának területén, mind a fordítás elméletének megteremtésére, mind a fordítóprogramok gyakorlati megvalósítására. Így alakult ki tehát egy tudomány, a matematikai nyelvészet gyakorlatibb aspektusú oldalága, a számítástechnikai nyelvészet.

## A formális nyelvek definíciója

A továbbiakban itt nyelv alatt mindig írott nyelv értendő. A nyelv mondatai ennek megfelelően mindig írott szövegek. Az írott szöveg alapegységei a karakterek. Ezek a szövegek egy adott véges szimbólum- illetve karakterkészlet elemeiből alkotott jelsorozatok.

## Fogalmak

Jelöljön  $V$  egy tetszőleges véges halmazt. Ezt a halmazt a nyelv **ábécéjének** nevezzük. Az ábécé elemeit betűknek hívjuk. Egy tetszőleges ábécé betűit az  $a, b, c$  szimbólumok jelölik.

A  $V$  ábécé karaktereiből alkotott sorozatok, vagyis  $V$ -beli szimbólumok sorozatai a  $V$ -feletti **szavak**. A  $V^i$  jelölés általában a  $V$  elemeiből alkotott, és pontosan  $i$  hosszúságú jelsorozatok halmazát jelöli.

Az  $i = 0$  eset a zérus hosszúságú, vagyis üres jelsorozatnak felel meg. Ennek jele  $\lambda$ .

Egy adott ábécéből képezhető összes lehetséges jelsorozatba természetesen beletartozik az üres jelsorozat, a  $\lambda$  is. Az üres sorozatot **üres szónak** nevezzük. Az üres szó egy matematikai absztrakció, ugyanis olyan szót jelent, melynek egyetlen betűje sincs.

Legyen  $V^*$  az a halmaz, amelynek elemei a  $V$  ábécéből (karakterkészletből) alkotott véges jelsorozatok halmaza. Így  $V^*$ -nak megszámlálhatóan végtelen eleme van. A  $V$  ábécé feletti összes szót tartalmazó nyelvet **teljes nyelv**-nek nevezzük, ezt jelöljük  $V^*$ -gal.

Az alkalmazott  $V^*$  jelöléssel, illetve jelölési konvencióval kapcsolatosan a következőket kell még tudnunk.

Egy  $V$  karakterkészletű  $L$  **nyelv** a  $V^*$  halmaznak egy tetszőleges részhalmaza.  $L \subseteq V^*$

Ha egy adott  $V^*$  halmazbeli  $v$  jelsorozat eleme az  $L$  nyelv által meghatározott részhalmaznak, akkor a  $v$  jelsorozat az  $L$  **nyelv szava**.

Egy **szó hosszán** a szót alkotó szimbólumok számát értjük, (esetleges többszöri előfordulással együtt) és ezt  $l(v)$ -vel jelöljük.

Jelölje  $v$  az  $a_1 \dots a_n$  és  $w$  a  $b_1 \dots b_m$   $V^*$ -beli szavakat. Ekkor  $l(v) = n$  és  $l(w) = m$ .

Például: Legyen az  $V$  ábécé két betűje  $a$  és  $b$ . Ekkor  $l(bab) = 3$

(a szó hosszát szokás még a következőképpen is jelölni:  $|bab|$  )

Egy adott jel (pl.  $y$ ) **előfordulásainak száma** egy adott  $u$  szóban:  $l_y(u)$ .

Például  $l_{\{a,b\}}(abcac) = 3$ .

**Részszo:**  $v$  részszoja  $u$ -nak, ha léteznek olyan  $w_1, w_2$  szavak, melyre  $u = w_1 v w_2$ .

Például:  $u = abbababa$   $u$ -nak a  $bab$  szó (valódi) rész-szoja.

Szó egy **prefixuma** (kezdőszelete vagy szókezdő részszoja):  $v$  az  $u$  szó prefixuma, ha van olyan  $w$  szó, hogy  $u = vw$ . Valódi prefixum, ha  $v \neq \lambda$ .

Például:  $u = abbababa$ . Ekkor  $u$ -nak 4 hosszúságú kezdő szelete:  $v = abba$

Szó egy **suffixuma** (végződése vagy záró részszoja):  $v$  az  $u$  szó suffixuma, ha van olyan  $w$  szó, hogy  $u = wv$ . Valódi szuffixum, ha  $v \neq \lambda$ .

Például:  $u = abbababa$ . Ekkor  $u$ -nak 4 hosszúságú végződése :  $v = baba$

Két szó **konkatenációja** (szorzata): A  $v = a_1 \dots a_n$  és  $w = b_1 \dots b_m$  szavak szorzatán (egyesítésén, összeláncolásán, konkatenációján) azt az új szót értjük, amely úgy jön létre, hogy  $w$ -t a  $v$  után írjuk. Az új szót  $vw$ -vel jelöljük.  $vw = a_1 \dots a_n b_1 \dots b_m$

Például: Legyen  $V = \{a, b\}$ . Ekkor az *abba* szó *baba* szóval való szorzata *abbababa* lesz (ami persze nem egyezik meg a *babaabba* szóval, mivel a szorzás általában nem kommutatív).

**Szó hatványa:** A  $v$  szó nemnegatív egész hatványai a  $v$  szó önmagával vett szorzatai  $v^0 = \lambda$ ,  $v^1 = v$ ,  $v^2 = vv$ , ...

Például:  $baba = (ba)^2$ .

**Szó megfordítása** (tükrözés): A  $v$  szó tükörképének azt a  $v_1$ -gyel jelölt szót nevezzük, amely a  $v$  betűit fordított sorrendben tartalmazza. Legyen  $v = a_1 \dots a_n$ , ekkor  $v$  megfordítása  $v_1 = a_n \dots a_1$ .

**Szófüggvények:** Legyen  $V_1$  és  $V_2$  két tetszőleges, nem feltétlenül különböző ábécé. A  $V_1^*$ -ot  $V_2^*$ -ra képező függvényeket szófüggvényeknek nevezzük.

### Elemi szóműveletek (szófüggvények)

- egy betű beszúrása a szóba, adott helyre;  
pl.  $bab \rightarrow baba$ ,  $kapa \rightarrow k\alpha pa$
- egy adott helyen lévő betű törlése a szóból;  
pl.  $igaz \rightarrow gaz$ ,  $kész \rightarrow k\acute{e}s$ ,  $kész \rightarrow k\acute{e}z$
- egy adott helyen lévő betű másikra való kicserélése;  
pl.  $kerek \rightarrow kérek$ ,  $l\acute{e}p \rightarrow kép$ ,  $l\acute{e}p \rightarrow láp$
- adott helyen lévő két szomszédos betű felcserélése  
pl.  $szák \rightarrow zsák$

Két szó távolságán azon elemi szóműveletek minimális számát értjük, amelyek alkalmazásával az egyik szóból előáll a másik.

## Formális rendszerek

**A formális rendszerek két alkotóeleme:**

$V$ , egy tetszőleges ábécé

$H$ , a helyettesítési szabályok halmaza ( $V$ -ből alkotható szópárok halmaza)

jelölése:  $x \rightarrow y$

**Levezetések:**

példa:  $H = \{ \lambda \rightarrow c, a \rightarrow aa, ab \rightarrow ba \}$

(üres szó helyett  $c$  bárhova helyettesíthető,  $ab$  helyett  $ba$  helyettesíthető,  $a$ -t bárhol szaporíthatjuk)

$abaa \rightarrow abcaa \rightarrow cabcaa \rightarrow cbacaa \rightarrow cbaacaa \rightarrow cbaacaaa$

Az egyik szóból a másik **közvetlenül levezethető**, ha van az egyiknek olyan részlete, amely részletet a helyettesítési szabályban lévő párjával helyettesítve a másik szót kapjuk.

jelölése:  $v \Rightarrow w$

Az egyik szóból a másik **levezethető**, ha van olyan szó sorozat, aminek az első szava azonos a kiinduló (egyik) szóval, utolsó szava azonos a másik szóval, és minden előző szóból közvetlenül levezethető a következő szó, vagy  $v = w$  (azaz 0 lépésben levezethető)

jelölése:  $v \Rightarrow^* w$

Szóproblémának nevezzük azt a kérdést, hogy egy adott formális rendszerben egy adott  $v$  szóból levezethető-e vagy sem egy adott  $w$  szó. A szóprobléma algoritmikusan nem eldönthető probléma.

**Formális rendszerek típusai:**

- **Algoritmusok**  
Markov algoritmus



- **Nyelvgenerálók**  
**Generatív rendszer**  
**Generatív grammatikák**

## Markov algoritmus

Legyen egy  $W(V, H, H_1)$  rendszerünk, ahol  $V$  egy ábécé,

$H$ , a helyettesítési szabályok rendezett halmaza,

$H_1$  pedig a záróhelyettesítések halmaza! (A  $H_1$  halmaz része a  $H$  halmaznak)

A Markov-algoritmus alkalmazható a  $v$  szóra, ha van olyan helyettesítési szabály, amelynek a bal oldala részszoja  $v$ -nek.

A szabályok közül az algoritmus során mindig a legelső alkalmazható szabályt kell alkalmazni a bal oldalról legelső előfordulásra.

Az algoritmus akkor áll meg, ha a helyettesítésünk záróhelyettesítés, vagy ha már nem alkalmazható rá egyetlen szabály sem.

Például egy levezetés:

Unáris-bináris átalakítás

Szabályok:

$xII$	$\rightarrow$	$Ix$
$xI$	$\rightarrow$	$1$
$Ix$	$\rightarrow$	$I0$
$x1$	$\rightarrow$	$.1$
$x$	$\rightarrow$	$.0$
	$\rightarrow$	$x$

Alkalmazott szabályok	Kapott karakterek
	IIIII
$\rightarrow x$	$xIIIII$
$xII \rightarrow Ix$	$IxIII$
$xII \rightarrow Ix$	$IIxI$
$xI \rightarrow 1$	$II1$
$\rightarrow x$	$xII1$
$xII \rightarrow Ix$	$Ix1$
$Ix \rightarrow I0$	$I01$
$\rightarrow x$	$xI01$
$x1 \rightarrow 1$	$101$

## Generatív rendszerek

(hova lehet eljutni az adott szabályokkal a kijelölt startszavakból)

**Három alkotóeleme van:**

V egy tetszőleges ábécé

Ax az axiómák (startszavak) halmaza

H a helyettesítési szabályok halmaza

Generatív rendszer által generált nyelv: olyan szavak halmaza, melyekhez tudunk olyan startszót találni, hogy belőle a szavak levezethetők legyenek. A nyelv szavait úgy kapjuk meg, hogy kiindulunk az egyik startszóból, majd a szabályokat tetszőlegesen alkalmazzuk.

**Generatív grammatikák**

Eddigiekben a nyelvek egyszerű definíciójával foglalkoztunk, egy formális nyelvet úgy definiáltunk, hogy mint halmazt felfogva, felsoroltuk az elemeit, a szavakat. Ezzel a módszerrel azonban csak kis elemszámú, véges nyelveket tudunk definiálni. A végtelen sok szót tartalmazó nyelvek azonban felsorolással nem adhatók meg. Volt már eddig is példa végtelen nyelv megadására, de a „generáló szabályt” egyszerű szöveges leírással adtuk.

A továbbiakban olyan módszert nézünk, ami a végtelen sok szót tartalmazó nyelv korrekt leírása: a nyelv szavait egy generáló nyelvtan segítségével fogjuk generálni.

**A generatív grammatika négy alkotóeleme:**

Nemterminális szimbólumok: egy másik ábécé, melynek jeleit csak segédeszközként használjuk a generálás során, betűi nem szerepelnek a generált nyelv szavaiban.

Terminális szimbólumok: egy ábécé, melynek elemeiből a szavak felépülnek (vagyis a definiálandó nyelv ábécéje).

Startszimbólum: ami tulajdonképpen egy rögzített nemterminális jel (egy kitüntetett szimbólum).

Helyettesítési szabályok, amelyekkel a szavak származtathatóak. A bal oldalukon van legalább egy nemterminális szimbólum.

példa: az  $\langle \{S,A\}, \{a, b\}, S, \{S \rightarrow aAb, aA \rightarrow aaAb, A \rightarrow \lambda\} \rangle$   
négyes egy generatív nyelvtan.

### Egy nyelv szavainak generálása a generatív grammatikával:

Kiindulunk a startszimbólumból mint egybetűs szóból.

Ezek után az éppen vizsgált szó valamelyik részszavát - valamilyen helyettesítési szabály alapján - egy másik szóra cseréljük ki.

Az eljárást akkor fejezzük be, ha olyan szóhoz jutunk, amely csak terminális jelekből áll.

példa: az  $\langle \{S,A\}, \{a, b\}, S, \{S \rightarrow aAb, aA \rightarrow aaAb, A \rightarrow \lambda\} \rangle$

példa: az  $\langle \{S,A\}, \{a, b\}, S, \{S \rightarrow aAb, aA \rightarrow aaAb, A \rightarrow \lambda\} \rangle$

példa: az

$\langle \{S,A\}, \{a, b\}, S, \{S \rightarrow aAb, aA \rightarrow aaAb, A \rightarrow \lambda\} \rangle$

generatív grammatika esetén a

$S \Rightarrow aAb \Rightarrow aaAbb \Rightarrow aaaAbbb \Rightarrow aaabbb$

levezetés érvényes.

## Chomsky nyelvosztályok

A nyelveket grammatikákkal jellemezhetjük. Noam Chomsky, a számítástechnikai nyelvészet legnagyobb egyénisége a nyelveket osztályokba sorolta az azokat generáló nyelvtanok helyettesítési szabályainak bonyolultsága szerint.

Chomsky négy nyelvosztályt definiált. Ezeket az osztályokat számokkal jelölte a következők szerint: van 0-s, 1-es, 2-es és 3-as nyelvosztály. Az osztály sorszámának növekedésével a generáló nyelvtan szabályai egyre szigorúbbak. Az egyes nyelvosztályok elemeit generáló nyelvtanok a következő helyettesítési szabályokat tartalmazzák:

### 0. nyelvosztály-**Mondatszerkezetű** nyelvek

nincs külön megkötés

Minden nyelvtanra, így erre is teljesülni kell, hogy a helyettesítési szabályok bal oldalán legalább egy nemterminálisnak lennie kell.

1. nyelvosztály-**Környezetfüggő** (Context Sensitive, CS) **nyelvek**

A helyettesítési szabályok alakját kétféle módon is jellemezhetjük:

$$\beta A \gamma \rightarrow \beta \alpha \gamma$$

Jelentése a következő: az  $A \rightarrow \alpha$  szabály csakis a  $\beta\gamma$  környezetben alkalmazható.

Erről az értelmezésről kapta a nyelvosztály a nevét.

Ugyanezen nyelvosztály egy másik elnevezése, meghatározási módja:

$$\alpha \rightarrow \beta, \text{ ahol } |\alpha| \leq |\beta|$$

A fenti megkötés azt fejezi ki, hogy a helyettesítési szabályok jobb oldala nem lehet rövidebb a bal oldalnál. Ezeket tehát nemcsökkentő nyelveknek is nevezik.

$$\alpha \rightarrow \beta, \text{ ahol } \beta \text{ jelsorozat nem lehet rövidebb az } \alpha\text{-nál}$$

2. nyelvosztály-**Környezetfüggetlen** (Context Free, CF) **nyelvek**

A helyettesítési szabályok alakja:

$$A \rightarrow \alpha$$

ahol  $\alpha$  tetszőleges, mind terminális, mind nemterminális szimbólumokat tartalmazható jelsorozat, vagyis egy adott  $A$  nemterminális szimbólum mindig helyettesíthető az  $\alpha$  jelsorozattal, függetlenül attól, mi a nemterminális környezete. A szabályok alkalmazása itt független tehát a környezettől, ezért kapta az osztály a környezetfüggetlen nyelvek nevet.

3. nyelvosztály-**Reguláris nyelvek** (Jobbreguláris)

Nyelvtanában csak kétféle helyettesítési szabálytípus megengedett:

$$A \rightarrow a, \text{ illetve } A \rightarrow aB$$

A helyettesítési szabály bal oldalán mindig egyetlen nemterminális, jobb oldalán pedig vagy egyetlen terminális szimbólum, vagy egy terminális és egy nemterminálisból álló jelsorozat áll. Az ilyen alakú nyelvtanokat jobbreguláris nyelvtanoknak nevezzük.

(Létezik balreguláris nyelvtan is, akkor a jobboldali szimbólumok felcserélődnek )

Különbséget kell tenni a nyelv és a nyelvtan között. Egy nyelvtanhoz egy és csakis egy nyelv tartozik, az a nyelv, amit az adott nyelvtan generál. Egy adott nyelvnek viszont több generáló nyelvtana is létezhet.

Egy adott nyelvtan akkor tesz eleget egy nyelvosztály követelményeinek, ha a nyelvtan valamennyi szabálya megfelel a feltételeknek.

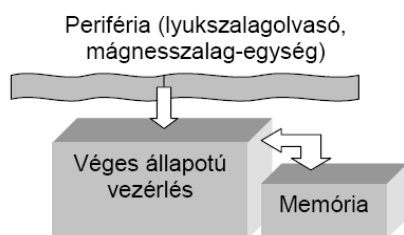
Mivel a nyelvosztály sorszámát növelve a helyettesítési szabályok korlátozásai súlyosbodnak, ezért az 1. típusú nyelvtan egyben 0. típusú is, és a 3. nyelvosztály nyelvtana a 2-es és 1-es nyelvosztály megkötéseit is kielégíti.

## Nyelvek és automaták

A tartalmazás problémája a számítástechnikai nyelvészet egyik alapvető kérdése.

Megfogalmazása a következő: ha adva van egy nyelv a grammatikájával és adva van egy jelsorozat, akkor el kell dönteni, hogy az adott jelsorozat eleme-e a megadott nyelvnek, generálható-e az adott jelsorozat az adott nyelvtannal. A matematikai nyelvészet erre a feladatra, a tartalmazás kérdésének megválaszolására új matematikai objektumokat definiált, az automatákat.

A különböző nyelvosztályokhoz különböző bonyolultságú nyelvtanok tartoznak, így az egyes nyelvosztályokhoz egyre bonyolultabb automata szükséges.



Az ábrán egy ilyen, matematikai automatát modellező fizikai automata elvi rajza látható. A négy különböző nyelvosztálynak négy különböző automata osztály felel meg.

Reguláris nyelvek	-	véges automaták
Környezetfüggetlen nyelvtanok	-	nemdeterminisztikus veremautomaták
Környezetfüggő-nemcsökkentő nyelvek	-	lineárisan korlátos automaták
0. típusú, mondatszerkezetű nyelvek	-	Turing-gépek

**Véges automaták:** csak olvasó egysége van, nincs memóriája, és csak véges számú állapota van. Az automata egy jelsorozat elemzését a következő módon végzi:

$q_0$  a kiindulási állapot,  $\delta$  a mozgási szabályok halmaza,  $F$  az elfogadó állapotok halmaza.

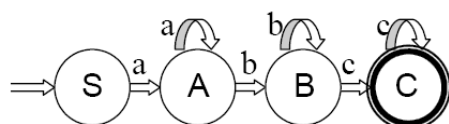
Az első karakter olvasásakor az automata a  $q_0$  kiinduló állapotban van. A mozgási szabályok mondják meg, hogy innen egy adott karakter olvasásakor milyen állapotba kerül át.

példa:  $\delta(A,b)=B$

**A** állapotban lévő automata **b** karaktert olvasva **B** állapotba kerül. Ha a változás megtörtént, az olvasófej továbblép a következő karakterre.

Az automata elfogad egy karaktersort, ha van olyan mozgássorozata, hogy végigolvasva a szöveget elfogadó állapotban áll meg.

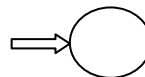
A véges automatákat gráfokkal ábrázolhatjuk, így működésüket szemléletessé tehetjük.



A gráf irányított, és csomópontjai megfelelnek az automata állapotainak. A mozgásokat az állapotok között a mozgáskor olvasott karakter nevét viselő élek jelölik.

Így például ha a mozgási szabály a következő:  $\delta(A, b)=B$ , akkor az A csomópontból a B csomópontba egy b jelű él vezet.

A kiindulási állapotot nyíllal jelöljük:



Az elfogadó állapotokat pedig kettős körrel:



Az automata nyelve azon jelsorozatok összessége lesz, amit az automata elfogad (vagyis a jelsorozat elolvasása után az automata elfogadó állapotba kerül)

A véges automaták azokat a nyelveket fogják elfogadni, amelyeket a reguláris nyelvtanok generálnak.

Az automaták működésének szemléltetésére feleltessünk meg az automata minden állapotának egy nemterminális szimbólumot (kezdőállapotnak a startszimbólumot)

Az automata minden mozgási szabályához rendeljünk egy levezetési szabályt, például:

$\delta(A,a) \rightarrow B$  mozgási szabályhoz a  $A \rightarrow aB$  levezetési szabályt.

A helyettesítési szabályok átírása mozgási szabályokká ugyanígy történik:

$A \rightarrow aB$  helyettesítési szabályból a  $\delta(A,a) \rightarrow B$  mozgási szabály lesz.

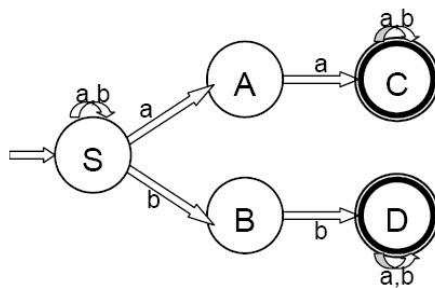
Az automatáknak és a megfelelő reguláris nyelvtanoknak az azonossága, (illetve az automaták nyelvének és a reguláris nyelvtanokkal generálható nyelveknek az azonossága igazolható.

### **Determinisztikus és nemdeterminisztikus véges automaták :**

Ha az automatánk minden állapot-karakter párjához legfeljebb egy mozgási szabály tartozik, akkor az automata determinisztikus. Ilyenkor egyértelműen tudjuk értelmezni az automata működését.

Ha azonban egy bizonyos állapot-karakter párhoz egynél több mozgási szabály is tartozhat, akkor az automata nemdeterminisztikus, ebben az esetben a működését követni már nem olyan egyértelmű. Valamennyi mozgáslehetőséget végig kell követnünk, mert nem tudjuk előre, melyik lesz az eredményes, melyik mozgássorozat vége lesz majd elfogadó állapot. Az automatákat nemdeterminisztikusság szerint egy számmal jellemezzük, a legtöbb mozgásiszabály számát adjuk meg annál az állapotnál, ami a „leggazdagabb”. Determinisztikus automatáknál ez a szám egy lesz, vagyis a determinisztikus automaták a

nemdeterminisztikusok speciális esetei.



1. nemdeterminisztikus automata

Két automata ekvivalens, ha ugyanazon jelsorozatokot fogadják el, s ugyanazokat utasítják vissza.

Bármely nemdeterminisztikus automatához létezik vele ekvivalens determinisztikus automata.



## Példák, feladatok

### Alapfogalmak

1. Legyen  $x=aba$ . Mi az értéke a következő kifejezésnek:  $x\lambda(b^3xa(ba)^0)$ ?
2. Hányszor fordul elő az  $aba$  szó az  $aababababbaba$  szóban?
3. Hány kezdőszelete és hány zárószelete van egy  $n$  hosszú szónak?
4. Hány különböző részszelete van az  $a^n b^n$  szónak?
5. Az  $y$  szó egyszerre kezdő és zárószelete az  $x$  szónak, és  $|x| < 2|y| < 2|x|$ .  
Mi jellemzi az  $x$  szót?
6. Adjuk meg azon  $x \in \{a; b\}^*$  szavakat, amelyekre  $ax = xa$  !
7. Mi az eredménye egy tetszőleges nyelv és a  $\{\lambda\}$  nyelv konkatenációjának?

### Formális rendszerek

#### példa:

Legyen adott egy ábécé:  $V = \{a, b, c\}$ .

Ekkor az ábécé karaktereiből alkotott jelsorozatok halmaza:

$$V^* = \{\lambda, a, b, c, aa, ab, ac, abb, ba \dots abcacbc \dots\}.$$

Ennek az alábbi részhalmaza legyen az  $L$  nyelv:

$$L = \{abc, aabbcc, aaabbbccc, \dots\}.$$

Az aabbcc jelsorozat az L nyelv egy szava. L az a nyelv, amelynek mondataiban megfelelő sorrendben ugyanannyi a, b, c karakter szerepel. Ezt a következőképpen is jelölhetjük:

$$L = a^i b^i c^i$$

1.feladat

Legyen adott egy ábécé  $V = \{a, b, 1\}$ , és egy nyelv L. Alkoss szavakat!

2. feladat

Dönts el az alábbi szavakról, részei-e a megadott nyelvnek...

$L = \{x \mid x \in \{a, b, c\}^* \text{ és } x\text{-ben kevesebb, mint 3 } a \text{ lehet}\}$

$\{\lambda, a, b, c, abc, aa, bb, cc, aaa, bbb, ccc, aabbcc, aaabbbccc, ababab, abcba, \dots\}$

1.feladat

Milyen helyettesítési szabályok segítségével lehet

- a) tetszőleges helyen álló a-t törölni,
- b) tetszőleges helyre bc-t beszúrni
- c) a csak a, b és c betűket tartalmazó szavakban a betűket betűrend szerint rendezni,
- d) a csak a, b és c betűket tartalmazó szavak betűit tetszőlegesen összekeverni,
- e) két a betű közé egy b-t beszúrni,
- f) két c betű közötti legfeljebb három a-t törölni,

## Markov algoritmus

példa:

Az  $(1^4 + 1^3)$  szóra alkalmazzuk az alábbi algoritmust!

$W(V, H, H1)$

$V = \{1, +\},$

$H = \{+1 \rightarrow 1+, 1+ \rightarrow 1\},$

$H1 = \{1+ \rightarrow 1\}.$

A lépésenkénti levezetés a következő:  $1111+111 \Rightarrow 11111+11 \Rightarrow$

$111111+1 \Rightarrow 1111111+ \Rightarrow 1111111.$

1. feladat

Adja meg azt a Markov-algoritmust, amely a csak a, b és c betűket tartalmazó szavakban a betűket betűrend szerint rendezi.

2. feladat

Adja meg azt a Markov-algoritmust, amely az  $n > 0$  pálcikából álló bemenethez előállít  $2n-1$  pálcikát.  $n=0$  esetén az algoritmus ne álljon meg.

3. feladat

Adja meg azt a Markov-algoritmust, amely az  $n$  pálcikából álló bemenethez előállít  $n^2$  pálcikát.

4. feladat

Adja meg azt a Markov-algoritmust, amely az  $n$  pálcikából álló bemenethez előállítja  $n$  bináris – felesleges 0-kat nem tartalmazó – alakját.

5. feladat

Adja meg azt a Markov-algoritmust, amely a csak a és b betűket tartalmazó szavakat tükrözi

6. feladat

Adja meg azt a Markov-algoritmust, amely a csak a és b betűket tartalmazó szavakban törli a szavakból az a betűket és megduplázza a b-ket.

7. feladat

Válogassuk ki a *bccaabcaab* szóban az *a*, *b*, *c*-ket!

A szabályok legyenek:  $ba \rightarrow ab$

$ca \rightarrow ac$

$cb \rightarrow bc$

8. feladat

Adjon meg olyan Markov algoritmust, amely egy bemenő bináris szó esetén annyi 1-est, majd annyi 0-ást ír, amennyi a bemenő szóban van!

Például: a 001011011 bemenetre a 111110000 szót adja eredményül.

## Generatív rendszerek

1. feladat

Adja meg az  $L = \{ a^n \mid n \text{ páratlan pozitív egész} \}$  nyelv generatív rendszerét.

2. feladat

Adja meg az  $L = \{ x \mid x \in \{a,b\}^* \text{ és } x \text{ páros sok } b\text{-t tartalmaz} \}$  nyelv generatív rendszerét.

3. feladat

Adja meg az  $L = \{ x, x+x, (x), x+x+x, (x)+x, x+(x), (x+x), ((x)), \dots \}$  nyelv generatív rendszerét.

4. feladat

Adja meg az  $L = \{ x \mid x \in \{a,b,c\}^* \text{ és } x\text{-ben a } b\text{-k egyetlen betű szigetet alkotnak} \}$  nyelv generatív rendszerét.

## Generatív grammatikák

pl. öttel osztható egész számok nyelve

$S \rightarrow 0|5|A0|A5 \quad A \rightarrow A1|A2|\dots|A9|A0|1|2|\dots|9$

1. feladat

Adjuk meg a következő nyelveket generáló jobblinéáris nyelvtanokat:

$$L = \{a; abba; bab; baba\}$$

$$L = \{w \in \{a; b; c\}^* \text{ és } w\text{-ben minden } a \text{ után áll egy } b\}$$

2. feladat

Milyen nyelvet generál az alábbi szabályrendszer?

$$S \rightarrow aB \quad A \rightarrow bAA$$

$$S \rightarrow bA \quad B \rightarrow bS$$

$$A \rightarrow aS \quad B \rightarrow b$$

$$A \rightarrow a \quad B \rightarrow aBB$$

3. feladat

Milyen nyelvet generál az alábbi szabályrendszer?

$$S \rightarrow aA \quad A \rightarrow aA \quad B \rightarrow bB \quad C \rightarrow cC$$

$$S \rightarrow bB \quad A \rightarrow bB \quad B \rightarrow cC \quad C \rightarrow aA$$

$$S \rightarrow cC \quad A \rightarrow \lambda \quad B \rightarrow \lambda \quad C \rightarrow$$

4. feladat

Készítsünk szabályrendszert, mely azokat a szavakat generálja, amelyek a-val kezdődnek és b-re végződnek.

(Jelölje A azt, hogy egy a-t kell generálnunk, C azt, hogy a-t vagy b-t generálhatunk, és B azt, hogy záró b-t kell előállítani. )

5. feladat

Készítsünk nyelvtant ahhoz az a,b betűk feletti nyelvhez, melynek szavai ugyanannyi a-t és b-t tartalmaznak.

6. feladat

Készítsünk nyelvtant ahhoz az a, b betűk feletti nyelvhez, melynek szavai megegyeznek a megfordításukkal (palindrómák)

7. feladat

Készítsünk nyelvtant mely az olyan a-kból álló szavakat fogadja el, melyek hossza 2 hatványa.

## Nyelvosztályok

### 1. feladat

Csoportosítsd a következő szabályokat aszerint, hogy melyik nyelvosztály feltételeinek felel meg.

$AbB \rightarrow AbbB$	$ac \rightarrow ca$	$B \rightarrow b$	$AaA \rightarrow a$
$A \rightarrow BA$	$B \rightarrow aB$	$a \rightarrow aA$	$bABa \rightarrow bAaBa$
$\lambda \rightarrow S$	$B \rightarrow ABA$	$AA \rightarrow b$	$bbB \rightarrow bbAbb$
$A \rightarrow abC$	$B \rightarrow cC$	$b \rightarrow B$	$abC \rightarrow AbC$
$S \rightarrow Saba$	$Ab \rightarrow bAa$	$AB \rightarrow aa$	$C \rightarrow CCC$

### 2. feladat

Adj példát olyan 0. típusú nyelvtani szabályra, amely nem 1. típusú!

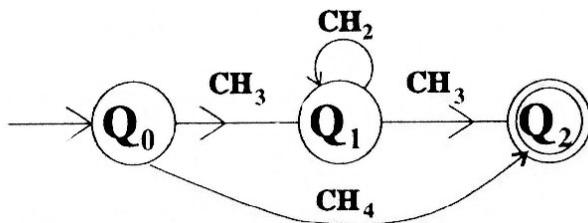
## Automaták

Elfogadó automata alkalmazása egy kémiai példával [Irodalomjegyzék: 8.]

Az automata megállapítja egy egyenesláncú, telített szénhidrogén vegyületről, hogy az a nem elágazó paraffinok csoportjába tartozik-e.

Ilyen vegyületek pl.  $\text{CH}_4$  (metán),  $\text{CH}_3\text{-CH}_3$  (etán),  $\text{CH}_3\text{-CH}_2\text{-CH}_3$  (propán)

Az automata az egyes molekula gyököket ( $\text{CH}_4$ ,  $\text{CH}_3$ ,  $\text{CH}_2$ ) mint bemenő karaktereket (terminálisokat) olvassa. Ha végig tudja olvasni a molekulát, és a végén elfogadó állapotban áll meg, akkor a molekula paraffin.





## Feladatok megoldásai

### Alapfogalmak

1. Legyen  $x=aba$ . Mi az értéke a következő kifejezésnek:  $x\lambda(b^3xa(ba)^0)$ ?  
A kifejezés értéke ababbbabaa.
2. Hányszor fordul elő az aba szó az aababababbaba szóban?  
4-szer fordul elő
3. Hány kezdőszelete és hány zárószelete van egy  $n$  hosszú szónak?  
 $n+1, n+1$
4. Hány különböző részszelete van az  $a^n b^n$  szónak?  
 $1+n+n^2$
5. Az  $y$  szó egyszerre kezdő és zárószelete az  $x$  szónak, és  $|x|<2|y|<2|x|$ .  
Mi jellemzi az  $x$  szót?  
 $x=z^n$ , ahol  $n>2$  és  $z\neq\lambda$ .

### Formális rendszerek

1.feladat

Milyen helyettesítési szabályok segítségével lehet

- a) tetszőleges helyen álló  $a$ -t törölni,  
 $a\rightarrow\lambda$
- b) tetszőleges helyre  $bc$ -t beszúrni  
 $\lambda\rightarrow bc$
- c) a csak  $a, b$  és  $c$  betűket tartalmazó szavakban a betűket betűrend szerint rendezni,  
 $ba\rightarrow ab, ca\rightarrow ac, cb\rightarrow bc$
- d) a csak  $a, b$  és  $c$  betűket tartalmazó szavak betűit tetszőlegesen összekeverni,  
 $ab\rightarrow ba, ba\rightarrow ab, ac\rightarrow ca, ca\rightarrow ac, bc\rightarrow cb, cb\rightarrow bc$
- e) két  $a$  betű közé egy  $b$ -t beszúrni,  
 $aa\rightarrow aba$



- f) két c betű közötti legfeljebb három a-t törölni,  
 $cac \rightarrow cc$ ,  $caac \rightarrow cc$ ,  $caaac \rightarrow cc$

## Markov algoritmus

### 1. feladat

Adja meg azt a Markov-algoritmust, amely a csak a, b és c betűket tartalmazó szavakban a betűket betűrend szerint rendezi.

A Markov-algoritmus programja:  $ba \rightarrow ab$ ,  $ca \rightarrow ac$ ,  $cb \rightarrow bc$

### 2. feladat

Adja meg azt a Markov-algoritmust, amely az  $n > 0$  pálcikából álló bemenethez előállít  $2n-1$  pálcikát.  $n=0$  esetén az algoritmus ne álljon meg.

A Markov-algoritmus programja:  $o| \rightarrow ||o$ ,  $|o \rightarrow .\lambda$ ,  $\lambda \rightarrow o$

### 3. feladat

Adja meg azt a Markov-algoritmust, amely az  $n$  pálcikából álló bemenethez előállít  $n^2$  pálcikát.

A Markov-algoritmus programja:  $o| \rightarrow pdo$ ,  $o \rightarrow x$ ,  $dp \rightarrow pd$ ,  $pd \rightarrow d|p$ ,  $px \rightarrow x$ ,  $dx \rightarrow x$ ,  $|x \rightarrow x|$ ,  $x \rightarrow .\lambda$ ,  $\lambda \rightarrow o$

### 4. feladat

Adja meg azt a Markov-algoritmust, amely az  $n$  pálcikából álló bemenethez előállítja  $n$  bináris – felesleges 0-kat nem tartalmazó – alakját.

A Markov-algoritmus programja:  $x|| \rightarrow |x$ ,  $x| \rightarrow 1$ ,  $|x \rightarrow |0$ ,  $x1 \rightarrow .1$ ,  $x \rightarrow .0$ ,  $\lambda \rightarrow$

### 5. feladat

Adja meg azt a Markov-algoritmust, amely a csak a és b betűket tartalmazó szavakat tükrözi

A Markov-algoritmus programja:  $\#a \rightarrow a\#$ ,  $\#b \rightarrow b\#$ ,  $xa \rightarrow xp$ ,  $xb \rightarrow xq$ ,  $po \rightarrow oa$ ,  $qo \rightarrow ob$ ,  $pa \rightarrow ap$ ,  $qa \rightarrow aq$ ,  $pb \rightarrow bp$ ,  $qb \rightarrow bq$ ,  $\# \rightarrow o$ ,  $xo \rightarrow .\lambda$ ,  $\lambda \rightarrow x\#$

## 6. feladat

Adja meg annak a Markov-algoritmusnak a szabályait, amely a csak a és b betűket tartalmazó szavakban törli a szavakból az a betűket és megduplázza a b-ket.

A Markov-algoritmus szabályai:  $a \rightarrow \lambda$ ,  $xb \rightarrow bbx$ ,  $x \rightarrow \lambda$ ,  $b \rightarrow xb$

## 7. feladat

Válogassuk ki a *bccaabcaab* szóban az *a*, *b*, *c*-ket!

A szabályok legyenek:  $ba \rightarrow ab$

$ca \rightarrow ac$

$cb \rightarrow bc$

bccaabcaab  
bcacabcaab  
baccabcaab  
abccabcaab  
abcaabcaab  
abaccbcaab  
aabcbcaab  
aabcbccaab

aabbcccaab  
aabbccacab  
aabbccaccab  
aabbaccab  
aabaaccab  
aaabbcccab  
aaabbccacb  
aaabbccacb

aaabbaccab  
aaaabaccab  
aaaabbcccb  
aaaabbccbc  
aaaabbcbcc  
aaaabbcbcc  
aaaabbcbcc

## 8. feladat

Adjon meg olyan Markov algoritmust, amely egy bemenő bináris szó esetén annyi 1-est, majd annyi 0-ást ír, amennyi a bemenő szóban van!

Például: a 001011011 bemenetre a 111110000 szót adja eredményül.

Szabály:  $01 \rightarrow 10$

## Generatív rendszerek

### 1. feladat

Adja meg az  $L = \{ a^n \mid n \text{ páratlan pozitív egész} \}$  nyelv generatív rendszerét.

A generatív rendszer axiómája:  $a$ , szabályai  $\lambda \rightarrow aa$ .

### 2. feladat

Adja meg az  $L = \{ x \mid x \in \{a,b\}^* \text{ és } x \text{ páros sok } b\text{-t tartalmaz} \}$  nyelv generatív rendszerét.

A generatív rendszer axiómája:  $\lambda$ , szabályai  $\lambda \rightarrow a$ ,  $\lambda \rightarrow bb$ .

3. feladat

Adja meg az  $L = \{ x, x+x, (x), x+x+x, (x)+x, x+(x), (x+x), ((x)), \dots \}$  nyelv generatív rendszerét.

A generatív rendszer axiómája:  $x$ , szabályai  $x \rightarrow x+x, x \rightarrow (x)$ .

4. feladat

Adja meg az  $L = \{ x \mid x \in \{a,b,c\}^* \text{ és } x\text{-ben a } b\text{-k egyetlen betű szigetet alkotnak} \}$  nyelv generatív rendszerét.

A generatív rendszer axiómája:  $b, ab, ba, aba$ , szabályai  $b \rightarrow bb, a \rightarrow aa, a \rightarrow c$ .

## Generatív grammatikák

1. feladat

Adjuk meg a következő nyelveket generáló jobblinéaris nyelvtanokat:

$$L = \{a; abba; bab; baba\}$$

$$L = \{w \in \{a; b; c\}^* \text{ és } w\text{-ben minden } a \text{ után áll egy } b\}$$

2. feladat

Milyen nyelvet generál az alábbi szabályrendszer?

$$S \rightarrow aB \quad A \rightarrow bAA$$

$$S \rightarrow bA \quad B \rightarrow bS$$

$$A \rightarrow aS \quad B \rightarrow b$$

$$A \rightarrow a \quad B \rightarrow aBB$$

$$S \Rightarrow aB \Rightarrow aBS \Rightarrow abaB \Rightarrow abab$$

$$S \Rightarrow aB \Rightarrow ab$$

$$S \Rightarrow aB \Rightarrow aaBB \Rightarrow aabB \Rightarrow aabb$$

$$S \Rightarrow aB \Rightarrow abS \Rightarrow abbA \Rightarrow abba$$

$$S \Rightarrow aB \Rightarrow abS \Rightarrow abbA \Rightarrow abbbAA \Rightarrow abbbbAAA \Rightarrow abbbbbaAA \Rightarrow abbbbbaaA \Rightarrow abbbbbaaa$$

.

.

.

vagyis  $L = \{x \mid x \in \{a,b\}^* \text{ és } x\text{-ben ugyanannyi } a \text{ van mint } b\}$

### 3. feladat

Milyen nyelvet generál az alábbi szabályrendszer?

$S \rightarrow aA$        $A \rightarrow aA$        $B \rightarrow bB$        $C \rightarrow cC$

$S \rightarrow bB$        $A \rightarrow bB$        $B \rightarrow cC$        $C \rightarrow aA$

$S \rightarrow cC$        $A \rightarrow \lambda$        $B \rightarrow \lambda$        $C \rightarrow \lambda$

$S \Rightarrow aA \Rightarrow a$

$S \Rightarrow aA \Rightarrow aaA \Rightarrow aaaA \Rightarrow aaa$

$S \Rightarrow bB \Rightarrow bbB \Rightarrow bb$

$S \Rightarrow bB \Rightarrow bcC \Rightarrow bc$

$S \Rightarrow cC \Rightarrow ccC \Rightarrow cc$

$S \Rightarrow cC \Rightarrow caA \Rightarrow caaA \Rightarrow caa$

vagyis akármennyi  $a, b, c$   $\{a, b, c, aa, bb, cc, aaa, bbb, ccc, bc, ca, caa, \dots\}$

### 4. feladat

Készítsünk szabályrendszert, mely azokat a szavakat generálja, amelyek  $a$ -val kezdődnek és  $b$ -re végződnek.

(Jelölje  $A$  azt, hogy egy  $a$ -t kell generálnunk,  $C$  azt, hogy  $a$ -t vagy  $b$ -t generálhatunk, és  $B$  azt, hogy záró  $b$ -t kell előállítani. )

$A \rightarrow aC$        $C \rightarrow B$

$C \rightarrow aC$        $B \rightarrow b$

$C \rightarrow bC$

## Nyelvosztályok

### 1. feladat

Csoportosítsd a következő szabályokat aszerint, hogy melyik nyelvosztály feltételeinek felel meg.

0. nyelvosztály:

$AaA \rightarrow a$

$AA \rightarrow b$

$abC \rightarrow AbC$

$Ab \rightarrow bAa$

1. nyelvosztály:  
 $AbB \rightarrow AbbB$   
 $bABa \rightarrow bAaBa$   
 $bbB \rightarrow bbAbb$   
 $AB \rightarrow aa$

2. nyelvosztály:  
 $A \rightarrow BA$   
 $B \rightarrow ABA$   
 $C \rightarrow CCC$   
 $S \rightarrow Saba$

3. nyelvosztály:  
 $B \rightarrow b$   
 $B \rightarrow aB$   
 $A \rightarrow abC$   
 $B \rightarrow cC$

nem szabály:  
 $ac \rightarrow ca$   
 $\lambda \rightarrow S$   
 $b \rightarrow B$   
 $a \rightarrow aA$

## 2. feladat

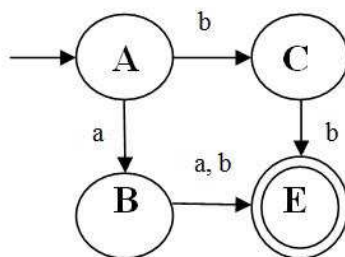
Adj példát olyan 0. típusú nyelvtani szabályra, amely nem 1. típusú!

pl.  $AB \rightarrow CD$

$AaA \rightarrow a$

## Automaták

Add meg a megadott nyelvtannal ekvivalens automatát:



## ***A formális nyelvek tanításának dokumentumai***

### **Formális nyelvek témakör elhelyezése a gimnáziumi tantervben**

#### **11-12. évfolyam**

Célok, feladatok, fejlesztendő területek:

Az informatika tantárgy tanításának alapvető célja felkelteni és ébren tartani a tanulók érdeklődését az informatika iránt, megismertetni eszközeit, módszereit. Ugyanolyan fontos cél annak az informatikai tudásnak, készségnek és képességnek a megszerzése, amelyek alkalmazásával más tantárgyak tanulását, későbbi tanulmányait, mindennapi életét, munkáját is segíteni képes.

Fontos kialakítani a tanulóban, a megszerzett ismeretek folyamatos bővítésének, az élethosszig tartó tanulásra ösztönözés igényét, hiszen az informatika, a tudomány legdinamikusabban fejlődő ága, eszközei módszerei rohamtempójú lépésben fejlődnek.

A tanuló ismerje a számítógépes munkakörnyezet munkavédelmi és ergonómiai szabályait, előírásait, a számítástechnikai eszközöket legyen képes céljuknak megfelelően használni. Legyen képes alkalmazni felhasználói szinten a számítógépet, a tanult szoftvereket, programokat, segédprogramokat, a számítógép perifériáit.

Legyen tudatában annak, hogy az általa megszerezhető informatikai tudás a mindennapi életben, a tanulásban, más tantárgyak tanulásában, a hasznára válik, ezért legyen motiváltabb a tantárgy tananyagának elsajátításában.

A tanuló legyen képes a különböző formákban megjelenő információkat felismerni, tudjon információt gyűjteni, információt különféle formákban megjeleníteni; szerezzen jártasságot az információk kezelésében. A megszerzett információit legyen képes felhasználni. Tegyen jártasságra szert a könyvtárak, elektronikus könyvtárak, különböző médiák valamint az internet használatában.

Ismerjen legalább egy operációs rendszert felhasználói szinten, tudja használni a segédprogramokat és azok szolgáltatásait. Ismerje a kommunikáció fogalmát, legyen képes a hálózat segítségével kapcsolatot kialakítani. A tanulóban alakuljon ki a korszerű alkalmazói készség, legyen képes kihasználni a számítógépek, az informatikai kultúra lehetőségeit.

**Fejlődjön a tanuló algoritmikus gondolkodás képessége. Ismerje meg az absztrakt gondolkodásmód jellegzetességeit.**

**Ismerje meg a számítástechnikai nyelvészet alapfogalmait, összefüggéseit, a formalizálás fontosságát, és lehetőségeit. Értse a tanuló, hogy ami egy formális nyelv eszközeivel kifejezhető, csak azt képes a számítógép értelmezni.**

**A tanuló tudja, hogy a problémamegoldáshoz alkalmazott programnyelvek speciális formális nyelvek, amelyek lehetővé teszik számukra a problémák informatikai módszerekkel való megoldását.**

**Tudja megkülönböztetni az egyes nyelvosztályokat, ismerje az osztályoknak megfeleltethető automata típusokat. Ismerje fel a formalizálás, automaták és a robotika összefüggéseit.**

Legyen képes különböző típusú dokumentumok, állományok szerkesztésére, módosítására, mentésére, archiválására, törlésére. Törekedjen az esztétikus formai megjelenítésre.

Ismerje az informatika és a társadalom kölcsönhatásait, ismerje a programok használatának jogi és etikai alapjait, az öncélú és túlzott informatikai eszközhasználat egészségkárosító, személyiségromboló hatásait.

Ismerje a kommunikációs lehetőségeket, a média szerepét a társadalomban.

Legyen képes különféle adatbázisokban keresni, azok szolgáltatásait igénybe venni.

Tudja igénybe venni a könyvtárak (hagyományos és elektronikus) szolgáltatásait, tudjon a könyvtárakban keresni.

Legyen nyitott a gyorsan változó informatikai eszközök, szoftverek fejlődési irányát felismerni és az informatikai tudást folyamatosan frissíteni.

## Ajánlás az éves óraszám felosztására

### 11. évfolyam

**Óraszám:** 56 óra/év  
1,5 óra/hét

<b>Témakör sorszáma</b>	<b>Témakör</b>	<b>Óraszám</b>
<b>1.</b>	Alkalmazói ismeretek	20 óra
<b>2.</b>	Infotechnológia	16 óra
<b>3.</b>	Infokommunikáció	6 óra
<b>4.</b>	Médiainformatika	8 óra
<b>5.</b>	Könyvtári informatika	4 óra
<b>6.</b>	Szabadon felhasználható	2 óra



## Ajánlás az éves óraszám felosztására

### 12. évfolyam

**Óraszám:**            **48 óra/év**  
**1,5 óra/hét**

<b>Témakör sorszáma</b>	<b>Témakör</b>	<b>Óraszám</b>
<b>1.</b>	Informatika-alkalmazói ismeretek	8 óra
<b>2.</b>	Infotechnológia	18 óra
<b>3.</b>	<b>Formális nyelvek és automaták</b>	<b>10 óra</b>
<b>4.</b>	Az információs társadalom	8 óra
<b>5.</b>	Szabadon felhasználható	4 óra

## Tanmenet részlet

Ajánlott óraszám: 10 óra

Óraszám	Ismeretek, fogalmak	Feladatok, szemléltetés
1.	A formális nyelvek és a formalizálás története, kialakulása, matematikai háttere	a formalizálás matematikai előzményei, szükségessége. Bemutató
2.	Szójátékoktól a formális nyelvekig ( a formális nyelvek alapfogalmai, definíciók) szóvizsgálatok	Szójátékok, szósorozatok, szóelemzések, szóműveletek.
3.	Formális rendszerek, Markov algoritmus, generatív rendszerek, levezethetőség, generatív grammatikák	Markov-algoritmus bemutató Feladatlapok-levezetések
4.	Nyelvtanok, egy nyelvtan elemei: terminális, nemterminális szimbólumok, startszimbólum, szabályok, példák a nyelvtanokra és a hozzátartozó nyelvekre	Adott nyelvtanokból szavak generálása
5.	Chomsky-féle nyelvosztályok, az egyes nyelvosztályok jellemzői, szabályai	Bemutató a nyelvosztályokról, feladatok, teszt
6.	Reguláris nyelvtanok, nyelvek, kifejezések	Feladatlap
7.	Automaták. Véges automaták. Az automaták formális definíciója. A véges automaták szerkezete. A működés leírása gráfokkal	Automaták működésének modellezése

8.	Nemdeterminisztikus és determinisztikus véges automáták. Elfogadó automata.	Mikor fogad el az automata egy jelsorozatot. Elfogadó automáták működésének modellezése.
9.	Reguláris nyelvek és a véges automáták megfeleltetése, a véges automáták által elfogadott nyelvek a reguláris nyelvek	Nyelvtanokból automáták, automátákból nyelvtanok felírása
10.	Egyéb nyelvosztályok automatái	Környezetfüggetlen nyelvek és a veremautomáták, nemcsökkentő nyelvek és a lineárisan korlátos automáták, 0-s nyelvosztály és a Turing-gép

## Óravázlat minta

Tanítás helye:	<b>Gimnázium</b>
Osztály:	<b>12. évfolyam</b>
Témakör:	<b>Formális nyelvek</b>
Tananyag:	<b>Szójátékoktól a formális nyelvekig</b>
Az óra típusa:	<b>Új ismeret szerző</b>
Didaktikai feladat:	<b>Új ismeret szerzése, gyakorlás, alkalmazó rögzítés</b>
Tanítás céljai:	

*Nevelési célok:* Logikus gondolkodásra nevelés. Algoritmikus gondolkodás fejlesztése. Önellenőrzésre nevelés. Szaknyelv pontos használatára nevelés.

*Oktatási célok:* Ismerje meg a formális nyelvek történetét. Értse meg a matematikai nyelvészet jelentőségét. Ismerje meg a formális nyelvek alapfogalmait, definícióit.

Alkalmazott munkaformák:

frontális osztálymunka, önálló munka, csoportmunka, közös megbeszélés, ellenőrzés,

Szemléltetés: számítógép, projektor, feladatlap

Fejlesztendő kompetenciák:

Digitális kompetenciák: korszerű alkalmazói készségek, digitális írástudás fejlesztése

Matematikai kompetenciák: matematikai ismeretek alkalmazása, felhasználása, logikus gondolkodás fejlesztése

Szociális kompetenciák: kooperatív képességek, együttműködési készségek fejlesztése a csoportmunka során

Szövegértési kompetencia: a feladatok helyes értelmezése

Idő	Az óra menete	Fogalmak, ismeretek, feladatok	Szakedidaktikai elemzés
0'	Tanóra eleji előkészületek		
1'	A mai órán egy új témakört kezdünk el, a formális nyelveket		<i>Célkitűzés</i>
2'	Bevezetesként nézzünk meg egy rövid bemutatót a formális nyelvek kialakulásának történetéről	Az előzmények elemzése, a fejlődés folyamata	Szemléltetés: PowerPoint bemutató
5'	Bevezető feladatok: <ol style="list-style-type: none"> <li>Jussunk el a KAS szóból betűhelyettesítésekkel a PÉK szóig úgy, hogy egyszerre csak egy betűt lehet helyettesíteni. (szólánc)</li> <li>Jussunk el a KAS szóból a PÉK szóig figyelembe véve azt a szabályt, hogy a K betű helyett csak P betűt lehet behelyettesíteni.</li> <li>Hova tudunk eljutni a LÉP szóból, ha a következő szabályok szerint léphetünk csak:  K→N, L→K, P→K, P→V</li> </ol>	<p>KAS→SAS→VAS→VÉS→ VÉR→KÉR→KÉP→PÉP→PÉK</p> <p>KAS→KÉS→KÉK→PÉK</p> <p>LÉP→LÉK→KÉK→KÉN</p> <p>LÉP→KÉP→NÉP→NÉV</p>	<p>Egyéni munka</p> <p>Algoritmikus gondolkodás fejlesztése</p>
10'	Formális nyelvek fogalma	A formális leírás azt jelenti, hogy valamilyen valós, vagy absztrakt rendszert a benne lévő összefüggésekkel együtt leírunk úgy, hogy matematikai szempontból kezelhetőek legyenek. Ennek egyik fő alkalmazása a formális nyelvészet	
12'	Alkossatok csoportokat!		
	Keressétek meg a következő fogalmak definícióit.	<ol style="list-style-type: none"> <li>csop. ábécé, szó, nyelv</li> <li>csop. előfordulás, résszó, kezdőszó, végződés</li> <li>csop. szó konkatenációja, (szorzata), szó hatványa, szó megfordítása</li> <li>csop. elemi szóműveletek</li> </ol>	<i>Csoportmunka</i>
	Hozzatok létre egy Alapfogalmak nevű új oldalt a Google csoportotokban és töltsétek fel oda a definíciókat,		

Idő	Az óra menete	Fogalmak, ismeretek, feladatok	Szakdidaktikai elemzés
22'	A csoportok képviselői mondják el a többieknek, mit jelentenek a kapott fogalmak. (csoportonként 2 perc)		<i>Közös megbeszélés és Csoportmunka</i>
30'	Oldjátok meg a kapott feladatlapokat (megoldási idő: 10 perc)	<p>Legyen <math>x=aba</math>. Mi az értéke a következő kifejezésnek: <math>x\lambda(b^3xa(ba)^0)</math>?</p> <p>Hányszor fordul elő az aba szó az aabababababa szóban?</p> <p>Hány kezdőszelete és hány zárószelete van egy <math>n</math> hosszú szónak?</p> <p>Hány különböző részszelete van az <math>a^n b^n</math> szónak?</p> <p>Az <math>y</math> szó egyszerre kezdő és zárószelete az <math>x</math> szónak, és <math> x  &lt; 2 y  &lt; 2 x </math>. Mi jellemzi az <math>x</math> szót?</p> <p>Adjuk meg azon <math>x \in \{a; b\}^*</math> szavakat, amelyekre <math>ax = xa</math> !</p> <p>Mi az eredménye egy tetszőleges nyelv és a <math>\{\lambda\}</math> nyelv konkatenációjának?</p>	<p>Szaknyelv pontos használatára nevelés</p> <p><i>Egyéni munka</i></p> <p>Algoritmikus gondolkodás fejlesztése</p> <p>Logikus gondolkodásra nevelés</p> <p>Fegyelmezett munkára nevelés</p>
40'	A helyes megoldások megbeszélése, értékelése		

## Tesztminta

### Alapfogalmak

?

Feleletválasztós teszt

Kit nevezünk a formális nyelvek atyjának?

☐ Kurt Gödel

☒ Noam Chomsky

☐ Alan Turing

[jó válasz](#)

Melyik évre tehető a formális nyelvek kislakulásának kezdete?

☐ 1950

☒ 1956

☐ 1965

[jó válasz](#)

Mivel jelöljük a teljes nyelvet?

☐  $V^+$

☐  $V_t$

☒  $V^*$

[jó válasz](#)

Mit értünk egy szó hosszán?

☐ A szót alkotó szimbólum fajták számát

☐ A szót alkotó szimbólumok számát többszöri előfordulások nélkül

☒ A szót alkotó szimbólumok számát többszöri előfordulásokkal együtt

[jó válasz](#)

A babbababa szónak melyik szó lesz prefixuma?

☒ bab

☐ baba

☐ abba

[jó válasz](#)

Mit jelent a konkatenáció?

☐ Szavak hatványozása

☒ Szavak összefűzése

☐ Szavak tükrözése

[jó válasz](#)

## **Irodalomjegyzék**

- 1. Bach Iván: Formális nyelvek**
- 2. Dömösi Pál, Fazekas Attila, Horváth Géza, Mecsei Zoltán: Formális Nyelvek és Automaták egyetemi jegyzet**
- 3. Dr. Várterész Magda: Formális nyelvek és automaták**
- 4. Révész György: Bevezetés a formális nyelvek elméletébe**
- 5. Demetrovics-Denev-Pavlov: A számítástudomány matematikai alapjai**
- 6. Dr. Hunyadvári László-Manhertz Tamás: Automaták és formális nyelvek**
- 7. Horváth Géza, Mecsei Zoltán, Nagy Benedek: Gyakorlati összefoglaló**
- 8. Kuki Ákos, Falucskai János, Dr. Tarnay Katalin: Bevezetés a formális nyelvek és automaták alkalmazásába**



## **A témával kapcsolatos Internet címek**

[http://people.inf.elte.hu/szlavi/InfoOkt/Mintak/Tanterv\\_tanmenetTGSzP.pdf](http://people.inf.elte.hu/szlavi/InfoOkt/Mintak/Tanterv_tanmenetTGSzP.pdf)

<http://www.kpszti.hu/kompetencia>

<http://www.inf.unideb.hu/~pappzol/hallginf/index.html>

<http://www.inf.unideb.hu/~varteres/>

<http://aszt.inf.elte.hu/~hunlaci/defi-1.pdf>

<http://aszt.inf.elte.hu/~hunlaci/book.pdf>

<http://aszt.inf.elte.hu/~hunlaci/gyakorlat2006t/gy1H.pdf>

<http://zeus.nyf.hu/~falu/formnyelv/>

<http://www.inf.u-szeged.hu/~gertom/Oktatas/Anyag/formnyelvek.php>

# A formális nyelvek története

## 1900 Párizs-első matematikai világkonferencia

Hilbert híres előadása, melynek hatására tág teret kapott az absztrakt gondolkodásmód széleskörű alkalmazása. Hilbert még úgy vélte, hogy létezik olyan univerzális módszer, melynek segítségével minden matematikai kérdés igaz vagy hamis volta eldönthető.



## 1910-es évek eleje

- Axel Thue  
az 1910-es évek  
elején kezdte  
tanulmányozni az  
adatsorozatokkal  
megadott utasítások  
tulajdonságainak  
vizsgálatát.



## 1930-as évek

- Kurt Gödel  
kimutatta, hogy  
létezik olyan  
precízen  
megfogalmazható  
matematikai állítás,  
melynek sem igaz,  
sem hamis volta nem  
bizonyítható.



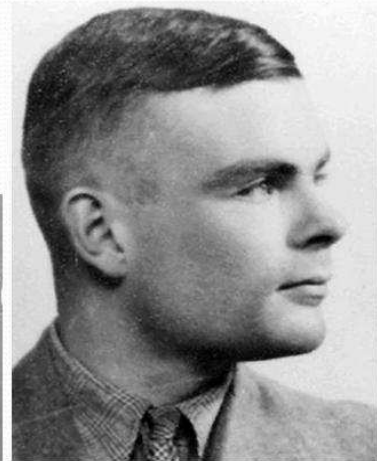
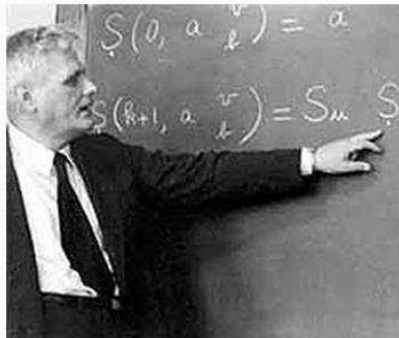


## 30-as évek második fele

Egy fiatal óriás, Alan Turing

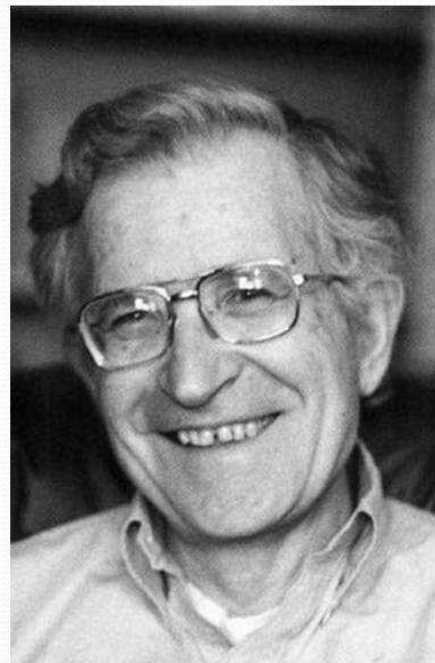
és vele párhuzamosan,  
Alonzo Church kapott  
hasonló módon meglepő  
eredményt.

Nevezetesen,  
hogynem  
létezik  
univerzális  
feladatmegoldó  
módszer.



## 1950-es évek

Noam Chomsky  
alapvető felfedezései  
jelentettek nagy  
lépéseket a formális  
rendszerek egy  
fontos típusa, a  
formális nyelvek  
matematikai  
elméletének  
területén



## Összefoglalás

Dolgozatom témájának olyan dolgot választottam, aminek Magyarországon nincs még gyakorlata, szakirodalma. Formális nyelvek tanítása középiskolában. Egy ilyen ötlet megvalósítása nagyon sokrétű feladat. Kevés támpontot adott két német tartomány tanmeneteinek átnézése, de itt is szinte csak címszavakba szedett utalást találtam. Szerettem volna egy olyan komplex tanítási környezetet kidolgozni, ami minden részletre kiterjed. Természetesen minden részből csak ötletszintű megvalósításra került sor, itt mindenhez csak egy-egy mozzanatot mutattam be, ennek a témának a teljes megvalósítása nem egy ilyen kiterjedésű dolgozat feladata.

A dolgozatom tehát nem kész munka, csak egy ötlet megvalósításának a kezdete, a továbbiakban szeretném ezt az ötletet és ezt a dolgozatot folyamatosan bővíteni. Olyan érdekes feladatokat szeretnék még gyűjteni, kitalálni hozzá, amik egyrészt felkeltik az érdeklődést a tanulóknak a formális nyelvek iránt, másrészt egyszerűvé és szemléletessé teszik ennek a korosztálynak a szintjén ezeket a nehéz összefüggéseket. Szeretném majd kidolgozni a téma összes órájának vázlatát, és minden témához korrekt mérési, értékelési rendszert. Az egész koncepció letanítását is tervezem, sajnos erre most nem volt lehetőségem. És végül szeretném majd ezeknek az óráknak a tapasztalatait is kiértékelni, tanulói méréseket végezni, és ezekből a tanulói mérésekből is levonni a következtetéseket.

## ***Köszönetnyilvánítás***

Szeretném megköszönni Falucskai Jánosnak, a Nyíregyházi Főiskola oktatójának, hogy egy új utat mutatott nekem, köszönöm a bizalmát és a biztatását, és köszönet dr. Papp Zoltánnak, a Debreceni Egyetem oktatójának, aki nagyon sokat segített abban, hogy tovább léphessek ezen az úton...